

Předmět KIV/TI - přednáška 6

# Úvod do teorie informace

Ing. Václav Vais, Ph.D.

[vais@kiv.zcu.cz](mailto:vais@kiv.zcu.cz)

# Pojem informace

- Norbert Wiener: Informace je název pro obsah toho, co si vyměňujeme s vnějším světem, když se mu přizpůsobujeme a působíme na něj svým přizpůsobováním.
- Informace = poznatky o prostředí, objektech, jevech a procesech v něm probíhajících.
- Informace snižuje nebo odstraňuje neurčitost „přijímacího systému“.
- Forma informace - text, obraz, řečový signál, .....
- Nosič informace – křídový prášek na tabuli, elektrický signál, optický signál, elektromagnetické vlnění, zmagnetizovaná doména na pevném disku, ...

# Matematická teorie informace

- Matematická teorie informace (C. E. Shannon, 1948) zkoumá informaci nezávisle na nosiči a odděleně od sémantického obsahu.
- Zpráva = posloupnost znaků, kterou odesílatel vybírá z předem dané abecedy, známé i příjemci.
- Jednotlivým znakům abecedy jsou přiřazeny významy, známé příjemci i odesílateli.
- Informace je poznatek, který zmenšuje nebo odstraňuje nejistotu týkající se výskytu určitého jevu ze známé množiny jevů.

# Neurčitost – motivační příklad

- Máme tři osudí, v každém je 10 lístků označených symboly A nebo B
  - 1. osudí - 5 lístků s A, 5 lístků s B
  - 2. osudí - 9 lístků s A, 1 lístek s B
  - 3. osudí - 10 lístků s A, žádný lístek s B
- Budeme intuitivně porovnávat „míru překvapení“ z výsledku toho, jaký lístek byl z každého z osudí vylosován
  - tah z 1. osudí - z výsledků A i B jsme „překvapení“ stejně
  - tah z 2. osudí - výsledek A moc nepřekvapí, o to více překvapí B (p-st jen 10%)
  - tah ze 3. osudí – překvapení žádné, výsledek předem jasný

# Neurčitost – motivační příklad

- Zřejmé – „míra překvapení“ z konkrétního výsledku souvisí s pravděpodobností, s jakou může jev nastat.
- Kvantifikovat „míru překvapení“ v případě nemožného výsledku nemá smysl (bylo by „nekonečně velké“).
- „Míru překvapení z výsledku náhodného jevu“ zavedeme jako seriózní veličinu – *elementární entropii písmene  $x_i$*  ; později ukážeme, že tato veličina souvisí s kvantifikací informace.

# Elementární entropie konkrétní realizace

- Elementární entropie  $H(x_i)$  je zřejmě funkcí pravděpodobnosti tohoto písmene  $H(x_i) = f(p(x_i))$  .
- Musí platit  $p_1 < p_2 \Rightarrow f(p_1) > f(p_2)$  ,  $H(x_i)$  je tedy funkcí klesající.
- V případě nezávislých jevů musí být elementární entropie aditivní, tedy  $f(p_1 \cdot p_2) = f(p_1) + f(p_2)$  (p-st toho, že současně nastanou dva nezávislé jevy je rovna součinu jejich p-stí) .
- Podmínkám vyhovuje  $f(x) = -\log(x)$  při libovolném základu větším než 1.
- Elementární entropie písmene  $x_i$  :  $H(x_i) = -\log_2 p(x_i)$  [bit]
- Jednotkou entropie je 1 bit (novější název – 1 Shannon, příliš se neujal).

# Střední entropie diskrétní náhodné veličiny

- Elementární entropie se vztahuje k jednomu konkrétnímu písmenu, p-sti ostatních písmen abecedy ji neovlivní.
- Střední entropie abecedy se vztahuje k celé abecedě, závisí na rozložení p-sti mezi všechna písmena. Je střední hodnotou elementárních entropií.
- Abecedu budeme chápat jako diskrétní náhodnou veličinu o  $r$  možných realizacích. Každé písmeno  $x_i$  má přiřazenu p-st  $p(x_i)$ , součet p-stí přes všechna písmena je roven 1.

- Střední entropie d. n. v.  $X$  : 
$$H(X) = - \sum_{i=1}^r p(x_i) \log_2 p(x_i)$$

Pro účely této definice  $p(x_i) = 0 \Rightarrow p(x_i) \cdot \log_2 p(x_i) \approx \lim_{x \rightarrow 0+} (x \cdot \log_2 x) = 0$

# Vlastností střední entropie $H(X)$

- Velikost střední entropie je omezena  $0 \leq H(X) \leq \log_2 r$  [bit]
- Mezní hodnoty:
  - $H(X) = 0$  jestliže může nastávat pouze jedna realizace  
pro  $p(x_i) = 0$  platí  $\lim_{x \rightarrow 0+} x \cdot \log_2 x = 0$   
pro  $p(x_i) = 1$  je  $\log_2 1 = 0$ ,  $\Rightarrow$  všechny členy v sumě jsou 0
  - $H(X) = \log_2 r$  jestliže všechny realizace mají stejnou p-st  $\frac{1}{r}$

$$\text{Pak } H(X) = - \sum_{i=1}^r \frac{1}{r} \log_2 \frac{1}{r} = - \log_2 \frac{1}{r} = \log_2 r$$



# Střední entropie - ilustrace

Návrat k motivačnímu příkladu:

Osudí 1:  $p(x_1) = 0,5, \quad p(x_2) = 0,5$

$$H(X) = -(0,5 \cdot \log_2 0,5 + 0,5 \cdot \log_2 0,5) = -\log_2 0,5 = -(-1) = 1 \text{ [bit]}$$

Osudí 2:  $p(x_1) = 0,9, \quad p(x_2) = 0,1$

$$\begin{aligned} H(X) &= -(0,9 \cdot \log_2 0,9 + 0,1 \cdot \log_2 0,1) = -[0,9 \cdot (-0,152) + 0,1 \cdot (-3,322)] = \\ &= -(-0,137 - 0,332) = 0,469 \text{ [bit]} \end{aligned}$$

Osudí 3:  $p(x_1) = 1, \quad p(x_2) = 0$

$$H(X) = -(1 \cdot \log_2 1 + 0 \cdot \log_2 0) = -(0 + 0) = 0 \text{ [bit]}$$

# Velikost informace jako rozdíl entropií

- Informace = veličina, která zmenšuje (v ideálním případě odstraňuje) neurčitost (entropii).
- Velikost informace, kterou přinesl náhodný jev = rozdíl neurčitosti ve sledované veličině **před** tím, než jev nastal, **a po** tom, co jev nastal.
- U motivačního příkladu (i u zdroje informace obecně) má smysl hledat odpovědi na dvě otázky:
  - a) kolik informace **přineslo** písmeno, které **bylo** vytaženo?
  - b) Kolik informace **může přinést** písmeno, které **bude** vytaženo

# Velikost informace jako rozdíl entropií

- V situaci ad a) kolik informace **přineslo** písmeno, které **bylo** vytaženo?

- známe tažené písmeno  $x_i$ , neurčitost **po** je nulová a platí

$$I(x_i) = H(x_i) - 0 = H(x_i), \text{ kde } I(x_i) = H(x_i) = -\log_2 p(x_i)$$

je elementární informace písmene  $x_i$ .

- V situaci ad b) Kolik informace **může přinést** písmeno, které **bude** taženo?

- nevíme, jaké písmeno  $x_i$  bude vytaženo. Nechceme-li z nějakého důvodu předjímat, jaké by **mělo být** taženo, vyjádříme očekávanou informaci jako střední hodnotu z  $I(x_i)$ , tedy  $I(X) = H(X)$ , kde  $I(X)$  je střední informace připadající na jedno písmeno.

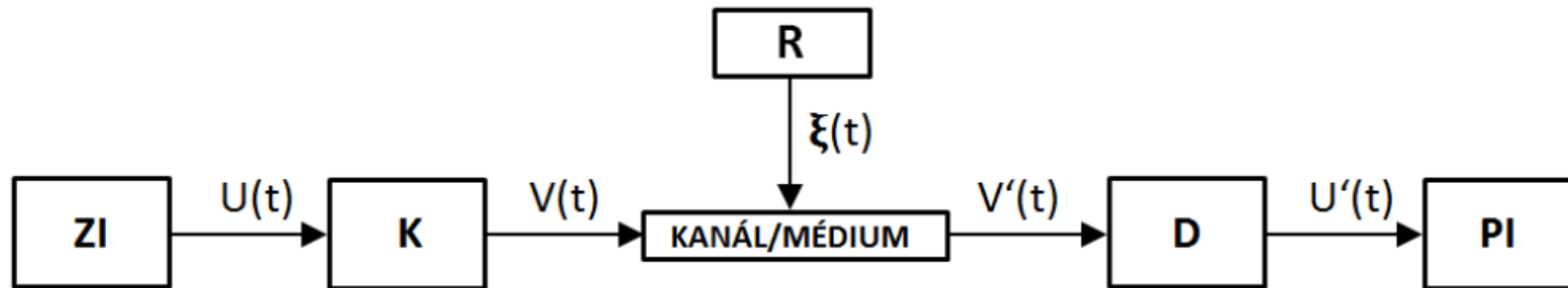
- Jednotkou elementární i střední informace je 1 bit (1 Shannon).

# Model sdělovací soustavy

- Cíle sdělování
  - přenést informaci v prostoru (přenos dat)
  - přenést informaci v čase (záznam dat na paměťové médium)
- Informace je nutné reprezentovat vhodnou fyzikální veličinou, která umožní dálkový přenos (elektrický nebo světelný signál, elektromagnetické vlnění, ....) nebo záznam na paměťové médium (elektrický náboj, zmagnetizovaná doména, prohlubeň nebo ploška na odrazné ploše optického disku, .....).
- **Informace proto musí být vhodným způsobem zakódována.**

# Model sdělovací soustavy

- Abstraktní model, vyhovuje úvahám o přenosu i záznamu informace



ZI ..... model zdroje informace

K ..... kodér

kanál/médium .....model sdělovacího nebo záznamového prostředí

D ..... dekodér

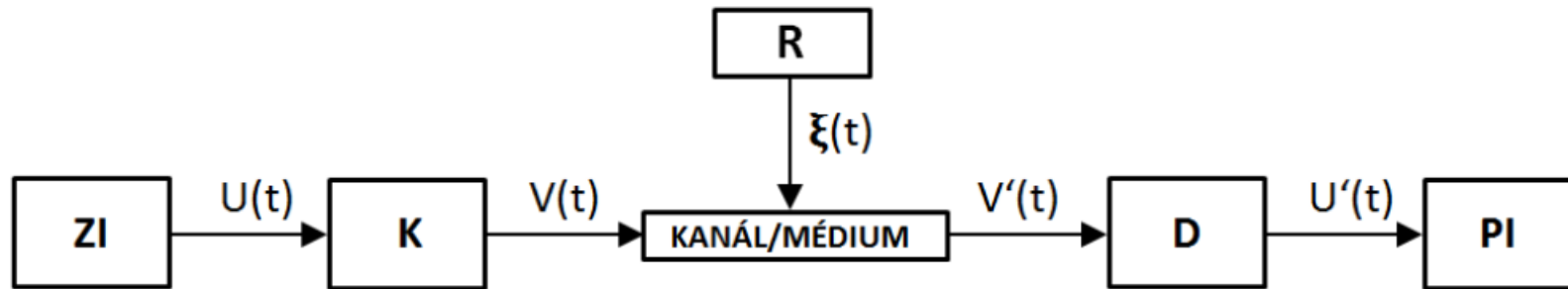
PI ..... příjemce informace

R ..... model rušení (nežádoucí vnější působení na sdělovací kanál/záznamové prostředí)

$U(t)$ ,  $V(t)$ ,  $V'(t)$ ,  $U'(t)$ ,  $\xi(t)$  ..... matematické modely průběhů příslušných signálů (obecně jsou to *náhodné procesy*)

# Model sdělovací soustavy

- Abstraktní model, vyhovuje úvahám o přenosu i záznamu informace



- Pouze v případě nulového rušení  $\xi(t)$  platí  $V'(t) = V(t)$  .
- Cílem přenosu/záznamu je, aby platilo  $U'(t) = U(t)$  .
- Součástí kodéru a dekodéru proto bývají i mechanismy pro eliminaci (nebo alespoň minimalizaci) důsledků rušení.

# Klasifikace zdrojů informace a kanálů

- Zdroj informace může být
  - diskrétní (generuje informaci v diskrétních časových okamžicích; zpráva je reprezentována řetězcem prvků nad abecedou zdroje)
  - spojitý (zpráva je reprezentována spojitou funkcí času)
- Sdělovací kanál může být
  - diskrétní (přenáší pouze znaky z nějaké konečné množiny)
  - spojitý (je schopen přenášet spojitý signál s charakteristikou v určitém omezeném rozsahu, např. frekvenční charakteristika)
- Funkce kodéru – transformovat zdrojové zprávy tak, aby byly přenositelné sdělovacím kanálem.

# Vztah mezi zdrojem informace a kanálem

- Diskrétní zdroj, diskrétní kanál – množina znaků zdroje a množina znaků kanálu nemusí být stejné, mohou mít různý počet znaků. Kodér řeší kódování znaků abecedy zdroje do řetězců abecedy kanálu. **Budeme řešit v kapitole Kódování.**
- Spojitý zdroj, spojitý kanál – frekvenční spektrum signálu zdroje nemusí odpovídat frekvenčnímu pásmu kanálu. Kodér řeší „přeložení frekvenčního pásma“, provádí spojitou analogovou modulaci signálu.
- Diskrétní zdroj, spojitý kanál – kodér řeší modulaci „hranatého signálu“, reprezentujícího posloupnost znaků generovanou zdrojem do frekvenčního pásma kanálu.

Příklad: modem pro připojení počítače na analogovou telefonní přípojku.



# Vztah mezi zdrojem informace a kanálem

- Spojitý zdroj, diskrétní kanál – kodér řeší vzorkování (v čase), kvantování (v úrovních) spojitého signálu a následné kódování vzorku.
  - Nyquistův–Shannonův vzorkovací teorém: Přesná rekonstrukce spojitého frekvenčně omezeného signálu z jeho vzorků je možná jen tehdy, pokud byla vzorkovací frekvence vyšší než dvojnásobek maximální frekvence obsažené ve spektru vzorkovaného signálu.
  - Počet úrovní, do kterých lze signál kvantovat, je pak omezen kapacitou kanálu.
  - Příklad praktického použití – pulzně kódová modulace PCM.

# Příklad kódování spojitého signálu do diskretního sdělovacího kanálu

**Příklad:** PCM pro přenos signálu z analogového telefonního přístroje diskretním sdělovacím kanálem s přenosovou rychlostí 64 kbit/s .

standardní frekvenční pásmo telefonního signálu

300 Hz – 3400 Hz

maximální frekvence obsažená v signálu

$f_{\max} = 3,4 \text{ kHz}$

dvojnásobek maximální frekvence

$2 \times f_{\max} = 6,8 \text{ kHz}$

frekvence vzorkování stanovená normou pro PCM

$f_{\text{vz}} = 8 \text{ kHz} > 2 \times f_{\max}$

perioda vzorkování

$T_{\text{vz}} = 125 \mu\text{s} = 1 / f_{\text{vz}}$

To znamená, že každých  $125 \mu\text{s}$  bude do sdělovacího kanálu odesílán jeden „zakódovaný vzorek“.

přenosová rychlost kanálu

$c = 64 \text{ kbit/s}$

počet bitů, které lze přenést v jedné periodě vzorkování

$n = 8 \text{ bitů} = c / f_{\text{vz}}$

počet úrovní, do kterých lze signál kvantizovat

$u = 256 = 2^n$

# Model diskrétního zdroje informace

- Diskrétní zdroj informace bez paměti = zdroj, kde vysílání jednotlivých znaků tvoří nezávislé jevy. To, jaký znak je vyslán v diskrétním časovém okamžiku  $T_n$  je statisticky nezávislé na tom, jaké znaky zdroj vyslal v okamžicích  $T_1$  až  $T_{n-1}$ .
- Takový zdroj lze popsat diskrétní náhodnou veličinou

$$X = \{x_1, x_2, \dots, x_r\} \text{ , } P(X) = (p(x_1), p(x_2), \dots, p(x_r)) \text{ , } \sum_{i=1}^r p(x_i) = 1$$

- Elementární entropii  $H(x_i)$  písmene  $x_i$  definujeme jako

$$H(x_i) = -\log_2 p(x_i)$$

# Model diskrétního zdroje informace

- Střední entropii  $H(X)$  zdroje  $X$  definujeme jako

$$H(X) = - \sum_{i=1}^r p(x_i) \log_2 p(x_i)$$

- Elementární informaci  $I(x_i)$  připadající na písmeno  $x_i$  definujeme jako

$$I(x_i) = H(x_i) = - \log_2 p(x_i)$$

- Informační vydatnost  $I(X)$  zdroje  $X$  definujeme jako

$$I(X) = H(X) = - \sum_{i=1}^r p(x_i) \log_2 p(x_i)$$

# Model diskrétního zdroje informace

- Jednotkou (jakékoli) entropie a informace je 1 bit (= 1 Shannon).
- Důležitou charakteristikou diskrétního zdroje informace je *redundance*  $\rho$

(nadbytečnost):

$$\rho = 1 - \frac{H(X)}{\log_2 r}$$

- Redundance je bezrozměrná veličina, často se udává v procentech.
- Velikost redundance je omezena:  $0 \leq \rho \leq 1$ .
- Mezních hodnot dosahuje tam, kde dosahuje mezních hodnot  $H(X)$  :
  - $\rho = 0$  tehdy, když je  $H(X) = \log_2 r$  ( $X$  má rovnoměrné rozložení)
  - $\rho = 1$  tehdy, když je  $H(X) = 0$  ( $X$  může nabývat jen jediné hodnoty)

# Redundance – ilustrační příklad

- Zdroj informace  $X = \{0, 1\}$  ,  $p(x_1) = 0,5$ ,  $p(x_2) = 0,5$  .
- Zpráva bude přenášena nespolehlivým kanálem s abecedou  $\{0, 1\}$ . Pro zvýšení pravděpodobnosti jejího správného vyhodnocení příjemcem bude každý znak „zakódován“ trojnásobným opakováním. Jaká je redundance zdroje, jaká je redundance po zakódování?
- Výpočet redundance zdroje:

$$H(X) = -(0,5 \cdot \log_2 0,5 + 0,5 \cdot \log_2 0,5) = -\log_2 0,5 = -(-1) = 1 \text{ [bit]}$$

$$\rho = 1 - \frac{H(X)}{\log_2 r} = 1 - \frac{1}{\log_2 2} = 1 - \frac{1}{1} = 0 \quad \text{Redundance zdroje je nulová.}$$

# Redundance – ilustrační příklad

- Výpočet redundance po zakódování
  - znaky jsou kódovány do trojic
  - takových trojic je celkem 8 (000, 001, 010, 011, 100, 101, 110, 111);  $r = 8$
  - zakódováním ale můžeme získat pouze dvě trojice (000, 111), obě s p-stí 0.5, pravděpodobnosti výskytu všech ostatních trojic na vstupu kanálu jsou nulové
  - tedy

$$H(X) = -(0,5 \cdot \log_2 0,5 + 0,5 \cdot \log_2 0,5) = -\log_2 0,5 = -(-1) = 1 \text{ [bit]}$$

$$\rho = 1 - \frac{H(X)}{\log_2 r} = 1 - \frac{1}{\log_2 8} = 1 - \frac{1}{3} = \frac{2}{3}$$

Soulad se selským rozumem (2 znaky ze 3 jsou nadbytečné)

Předmět KIV/TI - přednáška 6

# Úvod do kódování, kódy pro kanál bez šumu

Ing. Václav Vais, Ph.D.

[vais@kiv.zcu.cz](mailto:vais@kiv.zcu.cz)



# Kódování

- V dalších úvahách se budeme zabývat výhradně kódováním znakových řetězců pro přenos diskrétním sdělovacím kanálem.
- Primární důvod pro kódování – **přizpůsobení zdrojových řetězců vstupní abecedě kanálu.**
- Souběžně může kódování řešit i
  - efektivnější využití sdělovacího kanálu či paměťového média
  - zvýšení odolnosti přenášených či uložených zpráv proti rušení
  - šifrování zpráv

# Kódování

- Elementární příklad – kódy pro reprezentaci množiny textových a grafických znaků pomocí sedmibitových (osmibitových) značek z 0 a 1 (ASCII, EBCDIC, ....).  
(Umožňují uchovávat textovou informaci v dvoustavových paměťových prvcích počítače.)
- Teorie kódování = aplikace řady matematických disciplín
  - kombinatorika
  - lineární algebra
  - teorie čísel
  - Galoisova teorie těles
  - .....

# Kódování

- Kódování pro kanál bez šumu
  - předpoklad: kanál je „stoprocentně spolehlivý“
  - zvyšuje efektivitu využití kanálu/média snížením redundance ve zprávách
  - umožňuje bezztrátovou kompresi dat (**POZOR!** Nezaměňovat s kompresními metodami používanými u zvukových a obrazových záznamů).
- Kódování pro kanál se šumem
  - vkládají do zpráv redundantní informace
  - (za určitých pravděpodobnostních předpokladů) umožňují
    - detekci chyb (detekční kódy)
    - opravy chyb (korekční = samoopravné kódy)

# Kódování pro kanál bez šumu

Motivační příklad: Binární kódování dekadických číslic.  $A = \{0,1,2,3,4,5,6,7,8,9\}$  ..... abeceda zdroje  
 $B = \{0,1\}$  ..... (vstupní) abeceda kanálu

Zdrojový znak	Zakódovaný znak (kódová značka)			
	BCD 8421	BCD 2421	Excess 3	POSTNET 74910
0	0000	0000	0011	11000
1	0001	0001	0100	00011
2	0010	0010	0101	00101
3	0011	0011	0110	00110
4	0100	0100	0111	01001
5	0101	1011	1000	01010
6	0110	1100	1001	01100
7	0111	1101	1010	10001
8	1000	1110	1011	10010
9	1001	1111	1100	10100

# Kódování pro kanál bez šumu - příklad

Zdrojový znak	BCD 8421
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

- Všechny kódové značky mají stejnou délku, je to *blokový kód* (nemusí platit obecně)
- Kódování - „znak po znaku“ podle tabulky
- Zdrojový řetězec 3492
- Kódový řetězec 0011010010010010
- Dekódování - „rozsekat po čtveřicích“ a dekodovat „znak po znaku“ (lze jen u blokových kódů)
- Kódový řetězec 0011|0100|1001|0010
- Dekódovaný zdrojový řetězec 3492

# Kódování pro kanál bez šumu - příklad

	Kódová značka	
	Kód 1	Kód 2
0	0	0
1	0001	1000
2	1001	1001
3	0101	1010
4	1101	1011
5	0011	1100
6	1011	1101
7	0111	1110
8	01111	11110
9	11111	11111

- Kódové značky nemusí mít stejnou délku; může být výhodné, když p-sti výskytu písmen ve zdrojové zprávě nemají rovnoměrné rozložení.
- Kódování - „znak po znaku“ (podle tabulky Kód 1)
- Zdrojový řetězec 016058
- Kódový řetězec 0000110110001101111
- Délka kódového řetězce je 19 znaků.
- Kdybychom kódovali BCD, byla by délka 20 znaků.
- Jak dekódovat 0000110110001101111 ?
- Mechanickým rozsekáním na čtveřice to nepůjde.

# Kódování pro kanál bez šumu - příklad

- Pokusy o intuitivní dekódování („pokus – omyl“):

	Kódová značka	
	Kód 1	Kód 2
0	0	0
1	0001	1000
2	1001	1001
3	0101	1010
4	1101	1011
5	0011	1100
6	1011	1101
7	0111	1110
8	01111	11110
9	11111	11111

Pokus 1      0 | 0 | 0 | 0 | 1 1 0 1 | 1 0 0 0 1 1 1 1 1 1 0  
 0 0 0 0                      4    ?????????

Pokus 2      0 | 0 | 0 | 0 1 1 0 1 1 0 0 0 1 1 1 1 1 1 0  
 0 0 0    ?????????

Pokus 3      0 | 0 | 0 0 1 1 | 0 | 1 1 0 0 0 1 1 1 1 1 1 0  
 0 0                      5 0    ?????????

Pokus 4      0 | 0 0 0 1 | 1 0 1 1 | 0 | 0 | 0 | 1 1 1 1 1 | 1 0  
 0                      1                      6 0 0 0                      9    ???

.

.

Pokus x      0 | 0 0 0 1 | 1 0 1 1 | 0 | 0 0 1 1 | 1 1 1 1 0  
 0                      1                      6 0                      5                      8

- Proč vznikaly potíže?    Některé značky Kódu 1 jsou začátky jiných značek.

# Kódování pro kanál bez šumu - příklad

	Kódová značka	
	Kód 1	Kód 2
0	0	0
1	0001	1000
2	1001	1001
3	0101	1010
4	1101	1011
5	0011	1100
6	1011	1101
7	0111	1110
8	01111	11110
9	11111	11111

- Kódování - „znak po znaku“ (podle tabulky Kód 2)
- Zdrojový řetězec 016058
- Kódový řetězec 0100011010110011110
- Pokus o dekódování

Pokus 1      0 | 1 0 0 0 | 1 1 0 1 | 0 | 1 1 0 0 | 0 1 1 1 1  
                 0                      1                      6 0                      5                      8

- Nenastala situace, kdy by začátek dosud nezpracované části řetězce nebylo možné interpretovat jako kódovou značku; proto stačil jediný „pokus“.
- Proč? **Žádná kódová značka Kódu 2 není začátkem jiné kódové značky.**
- Na dekódování stačí Mealyho konečný automat.



# Kódování – základní definice

$A = \{a_1, a_2, \dots, a_r\}$  je *abeceda zdroje* (obsahuje  $r$  prvků)

$B = \{b_1, b_2, \dots, b_s\}$  je (vstupní) *abeceda kanálu (kódová abeceda)* (obsahuje  $s$  prvků)

$r > s$  (bez újmy na obecnosti).

*Kódováním znaků* rozumíme prosté (injektivní) zobrazení  $K : A \rightarrow B^+$

*Blokovým kódováním délky  $n$*  rozumíme prosté (injektivní) zobrazení  $K : A \rightarrow B^n$

*Kódováním řetězců (zpráv)* rozumíme zobrazení  $K^* : A^* \rightarrow B^*$ , jež je jednoznačně určeno kódováním znaků  $K$  takto

$$K^*(a_1 a_2 \dots a_l) = K(a_1) \cdot K(a_2) \cdot \dots \cdot K(a_l) \quad , \quad K^*(e) = e$$

*Podmínka jednoznačné dekódovatelnosti:* Kódování řetězců  $K^*$  je prosté zobrazení.

Každý blokový kód je jednoznačně dekódovatelný.

# Prefixové kódování

- *Prefixový kód* je kód, v němž žádná kódová značka není začátkem jiné kódové značky.
- Každý prefixový kód je jednoznačně dekódovatelný.
- Prefixové kódy lze dekódovat „znak po znaku“ v reálném čase přenosu, není třeba čekat na dokončení přenosu celé zprávy.
- K dekódování stačí konečný automat s výstupní funkcí Mealyho typu.
  - množinou vstupních symbolů automatu bude vstupní abeceda kanálu (kódová abeceda)
  - množinou výstupních symbolů bude zdrojová abeceda doplněná o neutrální symbol  $u$ .

# Prefixové kódování

- množina stavů automatu bude odpovídat prefixům (předponám) kódových značek s výjimkou kódových značek samotných
- počáteční stav automatu  $S$  odpovídá předponě  $e$
- přechodová funkce a výstupní funkce:
  - pokud je zpracován symbol, který doplní předponu reprezentovanou aktuálním stavem na kódovou značku, automat přejde do stavu  $S$  s tím, že bude vygenerován výstupní symbol odpovídající této značce
  - pokud doplněním aktuální předpony symbolem nevznikne kódová značka, automat přejde do stavu odpovídajícímu této nové předponě s tím, že na výstupu bude vygenerován neutrální symbol  $u$

# Dekódování prefixového kódu - příklad

	Kód 2 značka
0	0
1	1000
2	1001
3	1010
4	1011
5	1100
6	1101
7	1110
8	11110
9	11111

Stav	Vstupní znak	
	0	1
→ S	S/0	A/u
A	B/u	C/u
B	D/u	E/u
C	F/u	G/u
D	S/1	S/2
E	S/3	S/4
F	S/5	S/6
G	S/7	H/u
H	S/8	S/9

Prefix reprezen- tovaný stavem
<i>e</i>
1
10
11
100
101
110
111
1111

# Prefixové kódování

- Z podmínky na injektivnost zobrazení  $K$  vyplývá omezení vztahující se na délky kódových značek.

jinak řečeno

- Při zadaném počtu prvků abecedy zdroje a abecedy kódu nelze vytvořit „libovolně krátké“ kódové značky.
- Vztah mezi délkami kódových značek, počtem prvků zdrojové abecedy a počtem prvků kódové abecedy popisuje *Kraftova nerovnost*.
- Je-li nerovnost splněna, lze sestavit kód s požadovanými délkami značek, není-li nerovnost splněna, prefixové kódování s předpokládanými délkami značek neexistuje.

# Prefixové kódování

Ilustrační příklad: Navrhněte tříznakový prefixový kód, který umožní zakódovat šestiprvkovou zdrojovou abecedu tak, že dvě kódové značky budou mít délku 1 a zbylé čtyři značky budou mít délku 2. Tedy

$Z = \{A, B, C, D, E, F\}$  ..... abeceda zdroje

$X = \{0, 1, 2\}$  ..... kódová abeceda

Zdrojový znak	Požadovaná délka kódové značky	Pokus o konstrukci dle zadání	Korektní prefixový kód $K$
A	1	0	0
B	1	1	1
C	2	20	20
D	2	21	21
E	2	22	220
F	2	2?	221

# Kraftova nerovnost a McMillanova věta

- Kraftova nerovnost

$$s^{-d_1} + s^{-d_2} + \dots + s^{-d_r} \leq 1, \quad \text{kde } s \text{ je počet prvků kódové abecedy}$$

$d_i$  je délka  $i$  –té kódové značky  
 $r$  je počet prvků zdrojové abecedy

- Mc Millanova věta:

Každé jednoznačně dekódovatelné kódování splňuje Kraftovu nerovnost.

- Důsledek: Prefixové kódy lze chápat jako dostatečně obecnou a reprezentativní podtřídu jednoznačně dekódovatelných kódů (mají nejjednodušší mechanismus dekódování). Ke každému jednoznačně dekódovatelnému kódu existuje ekvivalentní prefixový kód.

# Kraftova nerovnost - příklad

(Návrat k ilustračnímu příkladu)

Pokus o konstrukci dle zadání
0
1
20
21
22
2?

$$s^{-d_1} + s^{-d_2} + s^{-d_3} + s^{-d_4} + s^{-d_5} + s^{-d_6} = 3^{-1} + 3^{-1} + 3^{-2} + 3^{-2} + 3^{-2} + 3^{-2} =$$

$$= \frac{1}{3} + \frac{1}{3} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} = \frac{3+3+1+1+1+1}{9} = \frac{10}{9}$$

Korektní prefixový kód K
0
1
20
21
220
221

$$s^{-d_1} + s^{-d_2} + s^{-d_3} + s^{-d_4} + s^{-d_5} + s^{-d_6} = 3^{-1} + 3^{-1} + 3^{-2} + 3^{-2} + 3^{-3} + 3^{-3} =$$

$$= \frac{1}{3} + \frac{1}{3} + \frac{1}{9} + \frac{1}{9} + \frac{1}{27} + \frac{1}{27} = \frac{9+9+3+3+1+1}{27} = \frac{26}{27}$$



# Kódy s minimální střední délkou značky

- Další krok v úvahách o prefixových kódech – formulace „kritérií kvality“, podle kterých vybereme optimální prefixový kód.
- Požadavek na efektivní využití kanálu  $\Rightarrow$  prostá úvaha – zdrojové znaky s větší  $p$ -stí výskytu ve zprávách kódovat značkami s menší délkou a naopak (viz Morseova abeceda).
- Exaktní ukazatel = střední délka kódové značky.
- Abychom ji byli schopni spočítat, musíme znát pravděpodobnostní rozložení znaků zdrojové abecedy.

# Kódy s minimální střední délkou značky

$$A = \{a_1, a_2, \dots, a_r\} , \quad P(A) = (p(a_1), p(a_2), \dots, p(a_r)) , \quad \sum_{i=1}^r p(a_i) = 1$$

*Střední délka kódové značky kódu  $K$  je pak definována jako*

$$\bar{d}(K) = \sum_{i=1}^r p(a_i) \cdot d(K(a_i))$$

$\bar{d}(K)$  .. střední délka kódové značky kódu  $K$

$a_i$  ... písmeno zdrojové abecedy

$p(a_i)$  ... pravděpodobnost výskytu písmena  $a_i$  zdrojové abecedy ve zprávách

$K(a_i)$  ... kódová značka příslušející písmenu  $a_i$  zdrojové abecedy

$d(K(a_i))$  . délka kódové značky příslušející písmenu  $a_i$

# Kódy s minimální střední délkou značky

- Takto definovaná střední délka kódové značky nám pro dané rozložení  $p$ -stí zdrojové abecedy umožňuje z množiny jednoznačně dekódovatelných kódů vybrat ten s minimální střední délkou.
- Jím kódované zprávy zaberou ze všech kódů nejmenší kapacitu sdělovacího kanálu nebo paměti.
- Tento princip využívají metody bezeztrátové statistické komprese dat.
- Jak kód s minimální střední délkou kódové značky zkonstruovat?
- Huffmanův algoritmus (1952)

# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

**Vstupy algoritmu:** zdrojová abeceda  $A$  o  $r$  prvcích a její pravděpodobnostní rozložení  $P(A)$   
kódová abeceda  $B$  o  $s$  prvcích

**Výstup algoritmu:** prefixové kódování  $K : A \rightarrow B^+$  takové, že  $\bar{d}(K)$  je minimální (tj. neexistuje jiné kódování  $\tilde{K}$ , pro které by platilo  $\bar{d}(\tilde{K}) < \bar{d}(K)$ )

## Algoritmus:

1. Prvky zdrojové abecedy  $A$  seřadíme podle jejich pravděpodobností  $p(a_i)$  do nerostoucí posloupnosti.
2. Takto seřazené prvky rozdělíme do skupin. Začínáme od prvků s nejvyšší pravděpodobností. Skupiny budou mít  $s - 1$  prvků. Výjimkou může být poslední skupina, která může mít od 2 do  $s$  prvků.
3. Sdružíme prvky v poslední skupině a nahradíme je *sduženou skupinou*, kterou zařadíme podle její součtové pravděpodobnosti na správné místo do posloupnosti. Rozdělení provedené v bodě 2 v tuto chvíli zaniká.

# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

4. Sdružíme posledních  $s$  prvků v posloupnosti a nahradíme je sdruženou skupinou, kterou zařadíme podle její součtové pravděpodobnosti na správné místo do posloupnosti.
  5. Bod 4 opakujeme tak dlouho, dokud nezískáme jedinou sdruženou skupinu se součtem 1.
  6. Zpětným chodem po větvích s —árního stromu vytvořeného v bodech 3 až 5 přiřadíme kódové značky listům stromu, tj. znakům zdrojové abecedy.
- Při tomto kódování dochází k redukci redundance ve zprávách

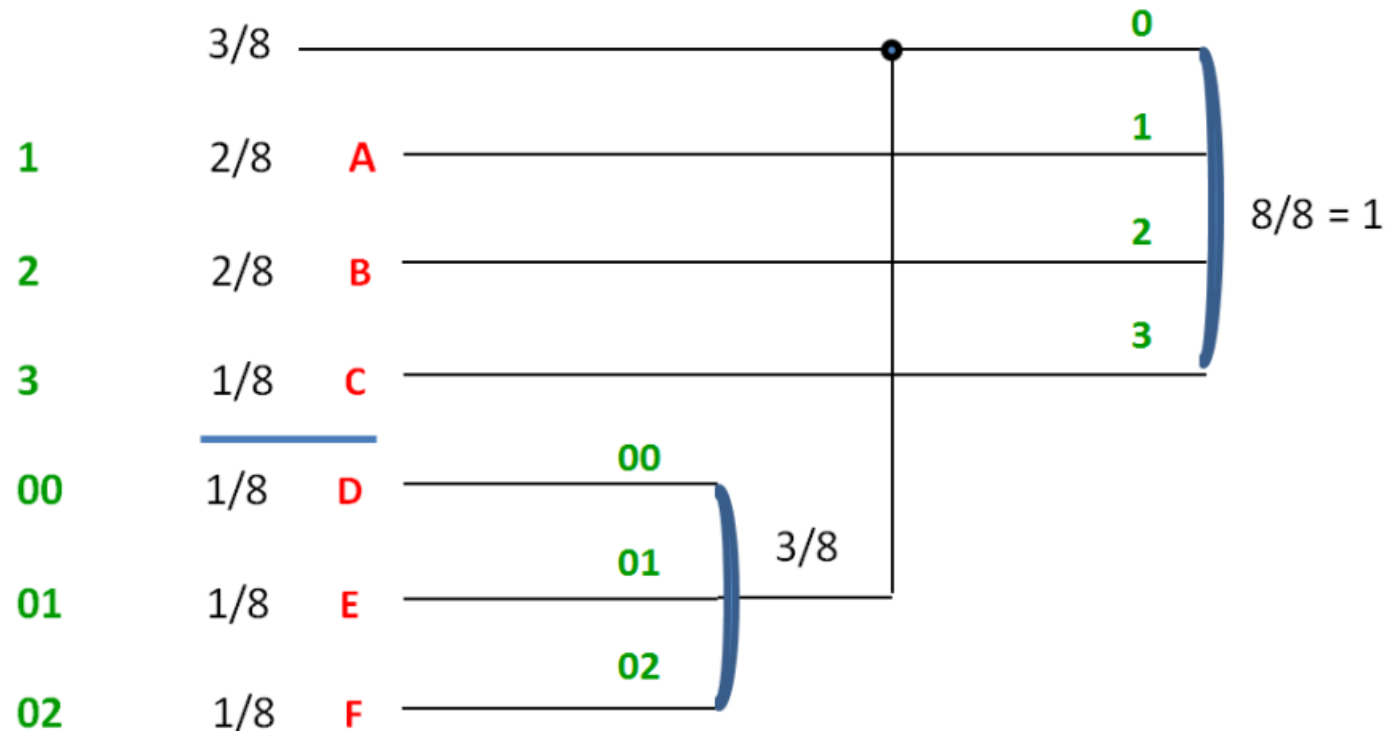
jinak řečeno

Redundance po zakódování je menší, než redundance ve zdrojové abecedě.

# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

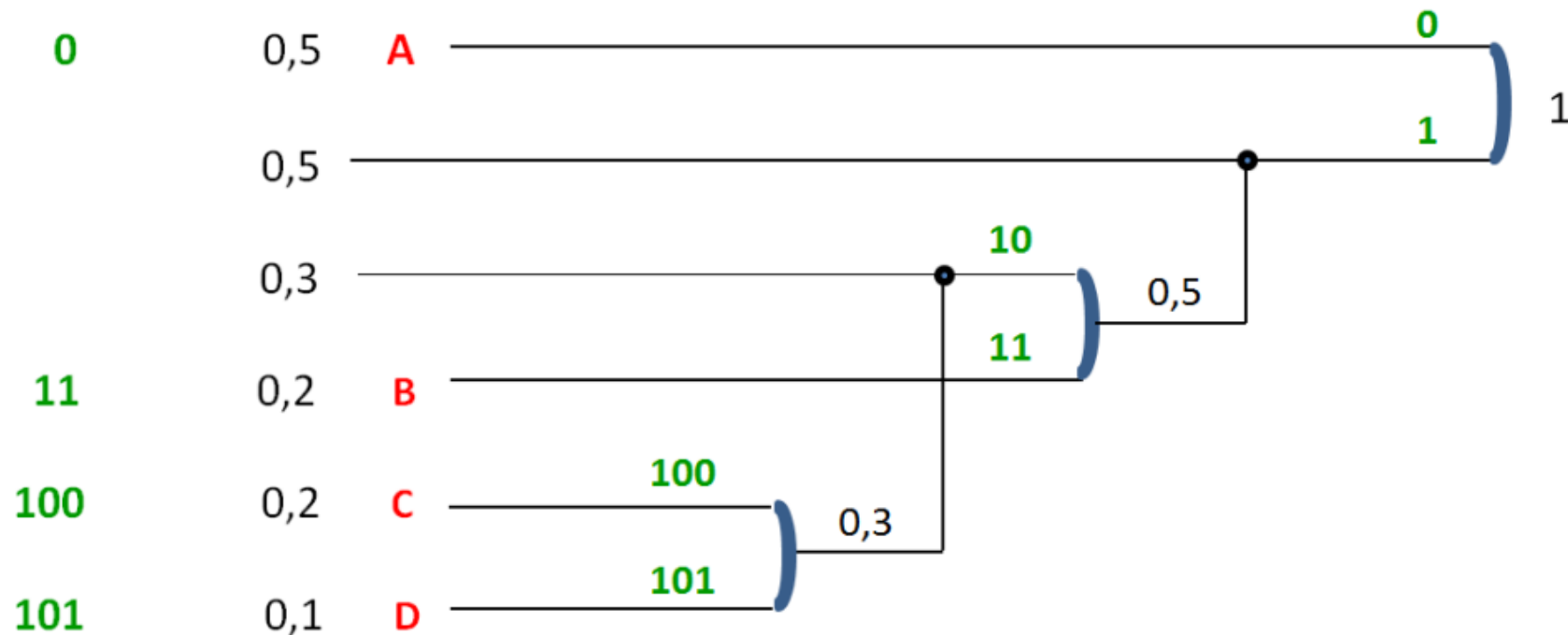
Ilustrační příklad: Do čtyřprvkové abecedy  $X = \{0,1,2,3\}$  zakódujte zdrojovou abecedu

$Z = \{A, B, C, D, E, F\}$  s rozložením  $P(Z) = \left(\frac{2}{8}, \frac{2}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}\right)$ .



# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

Ilustrační příklad: Do dvouprvkové abecedy  $X = \{0,1\}$  zakódujte zdrojovou abecedu  $Z = \{A, B, C, D\}$  s rozložením  $P(Z) = (0,5, 0,2, 0,2, 0,1)$  .



# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

Pro výpočet pravděpodobnostního rozložení kódové abecedy vyjádříme  $N_0$  (střední počet znaků 0 ve značce),  $N_1$  (střední počet znaků 1 ve značce) a střední délku značky  $\bar{d}$ .

Zdrojový znak	P-st znaku	Kódová značka	Délka kódové značky	Počet znaků	
				0	1
A	0,5	0	1	1	0
B	0,2	11	2	0	2
C	0,2	100	3	2	1
D	0,1	101	3	1	2

$$\bar{d} = 0,5 \cdot 1 + 0,2 \cdot 2 + 0,2 \cdot 3 + 0,1 \cdot 3 = 1,8$$

$$N_0 = 0,5 \cdot 1 + 0,2 \cdot 0 + 0,2 \cdot 2 + 0,1 \cdot 1 = 1,0 \quad p_0 = \frac{N_0}{\bar{d}} = \frac{1,0}{1,8} = \frac{5}{9}$$

$$N_1 = 0,5 \cdot 0 + 0,2 \cdot 2 + 0,2 \cdot 1 + 0,1 \cdot 2 = 0,8 \quad p_1 = \frac{N_1}{\bar{d}} = \frac{0,8}{1,8} = \frac{4}{9}$$



# Huffmanova konstrukce prefixového kódu s minimální střední délkou kódové značky

Nejprve spočítáme střední entropii a redundanci ve zdrojové abecedě:

$$H(Z) = -(0,5 \cdot \log_2 0,5 + 0,2 \cdot \log_2 0,2 + 0,2 \cdot \log_2 0,2 + 0,1 \cdot \log_2 0,1) = 1,76$$

$$\rho(Z) = 1 - \frac{1,76}{\log_2 4} = 1 - \frac{1,76}{2} = 0,12$$

Nyní již můžeme spočítat střední entropii a redundanci v kódové abecedě:

$$H(X) = -\left(\frac{5}{9} \cdot \log_2 \frac{5}{9} + \frac{4}{9} \cdot \log_2 \frac{4}{9}\right) = 0,99 \quad \rho(X) = 1 - \frac{0,99}{\log_2 2} = 1 - \frac{0,99}{1} = 0,01$$

Redundance 12% ve zdrojových zprávách se tedy Huffmanovým kódováním zredukovala na pouhé 1%.